Research Article

Nano Biomed Eng

EfficientNet - XGBoost: An Effective White-Blood-Cell Segmentation and Classification Framework

Bairaboina Sai Samba SivaRao, Battula Srinivasa Rao
School of Computer Science & Engineering, VIT-AP University, Amaravathi, India

Corresponding author. sreenivas.battula@gmail.com

Received: Oct. 20, 2022; **Revised:** Jan. 26, 2023; **Accepted:** Apr. 13, 2023

Citation: B.S.S. SivaRao, B.S. Rao. EfficientNet - XGBoost: an effective white-blood-cell segmentation and classification framework. *Nano Biomedicine and Engineering*, 2023.

http://doi.org/10.26599/NBE.2023.9290014

Abstract

In the human body, white blood cells (WBCs) are crucial immune cells that help in the early detection of a variety of illnesses. Determination of the number of WBCs can be used to diagnose conditions such as hematological, immunological, and autoimmune diseases, as well as AIDS and leukemia. However, the conventional method of classifying and counting WBCs is time-consuming, laborious, and potentially erroneous. Therefore, this paper presents a computer-assisted automated method for recognizing and detecting WBC categories from blood images. Initially, the blood cell image is preprocessed and then segmented using an effective deep learning architecture called SegNet. Then, the important features are devised and extracted using the EfficientNet architecture. Finally, the WBCs are categorized into four different types using the XGBoost classifier: neutrophils, eosinophils, monocytes, and lymphocytes. The advantages of SegNet, EfficientNet, and XGBoost make the proposed model more robust and achieve a more efficient classification of the WBCs. The BCCD dataset is used to evaluate the performance of the proposed methodology, and the findings are compared to existing state-of-the-art approaches based on accuracy, precision, sensitivity, specificity, and F1-score. Evaluation results show that the proposed approach has a higher rank-1 accuracy of 99.02% and outperformed other existing techniques.

Keywords: deep learning; SegNet; classification; white blood cell; EfficientNet; segment

Introduction

Blood is an essential component of the human body. It constitutes around 7% of total body weight in younger people. It is made up of 55% plasma, allowing it to circulate easily all over the body via the arteries [1–3]. The cells in the blood are divided into three types, which differ in their color, size, texture, morphology, and composition: thrombocytes (platelets), leukocytes (white blood cells, WBCs), and erythrocytes (red blood cells, RBCs). When examined under a microscope, these components have different morphologies and sizes, with WBCs being larger than the other cells due to them having both cytoplasm and nucleus. WBCs play significant roles in the immune system and are also known as immune cells. WBCs protect the body from infectious diseases and external invaders. WBCs have typically been classified into two categories [4–6]: nongranular and granular cells. The granular cells include basophils, eosinophils, and neutrophils, while the nongranular cells include lymphocytes and monocytes.

Neutrophils feature multilobed nuclei have two to five lobes each. They are the most prevalent

phagocytic cells, accounting for 50%–60% of all WBCs [7–9]. Meanwhile, eosinophils account for 1%–6% of all WBCs, with a bilobed nucleus in most cases. In the blood and bone marrow, basophils are one of the smallest WBC subsets, accounting for less than 2% of the total. Moreover, monocytes are a type of WBCs with nongranular cytoplasm that are formed in the bone marrow and account for 2%–10% of all WBCs [10–12]. T cells and B cells are the two types of lymphocytes, which are the smallest of the WBCs. They contain minimal cytoplasm and account for only 20%–30% of WBCs.

In healthy humans, these WBC subsets make up set proportions of total WBCs. Many disorders, such as bacterial infection and inflammation, are associated with abnormalities of the proportions of these WBC subpopulations [13–15]. As such, WBC classification and determination of the overall composition of WBCs are vital in medical diagnosis. Doctors regularly use such fundamental information for determining the severity of hematological diseases. Research on WBC classification has thus become an essential part of medical evaluation. Peripheral blood samples are sometimes examined manually by a hematologist, who examines such samples under a microscope. However, this technique is time-consuming and can lead to incorrect results because of fatigue or human error [16, 17]. Meanwhile, automated hematological analyzer machines (such as Sysmex) are prohibitively expensive, particularly in low-income nations. Against this background, there is a need for automated procedures to classify WBCs in peripheral blood smears. Automatic WBC classification techniques have long been a focus of researchers [18–20]. For example, a convolutional neural network (CNN)-based WBC classification technique was introduced by Yao et al [21]. Classification, feature extraction, segmentation, and image preprocessing phases were included in this framework. For image segmentation, region growing- and Otsu segmentation-based hybrid techniques have been used. Subsequently, Two-DCNN was implemented to perform feature extraction and classification.

Similarly, Cheuque et al. [22] implemented a multi-level CNN technique to separate polymorphonuclear and mononuclear cells, while for determining the region of interest in WBCs, the faster R-CNN technique was used. Two corresponding CNNs were later implemented to categorize the WBC classes. These networks were based on the model of MobileNet with a depth-wise separable convolution unit, with this unit being used for feature extraction from each channel. In addition, Çınar and Tuncer [23] proposed a hybrid Alexnet-Googlenet-SVM-based technique to categorize different kinds of WBCs with steps involving preprocessing, filtering, feature extraction, and classification. To segment the WBC nucleus, Lu et al. [24] developed a deep learning technique called WBC-Net with two hybrid techniques named Resnet and UNet++. Similarly, Tarek et al. [25] presented a three-phase segmentation and classification framework. In that study, the authors segmented the WBCs by multi-level thresholding. At this stage, the optimal threshold values were selected by the butterfly optimization algorithm. Subsequently, the feature extraction process was conducted to extract the shape and geometric features. Then, the classification was performed using the multilayer perceptron, which classifies blood cells into five categories, and the optimal bias and weights of the network were initialized by the Gray Wolf Optimization algorithm. Another study conducted by Tavakoli et al. [26] performed the segmentation and classification of WBCs based on Otsu thresholding and support vector machine, respectively. Based on the canonical correlation analysis, Patil et al. [27] combined the LSTM and CNN framework. In addition, in 2022, the existing methods for evaluating blood smear images and identifying leukemia from them were thoroughly reviewed by Mittal et al [28]. They bridged a gap in the literature by providing a thorough analysis of 149 papers outlining the techniques for analyzing blood smear images and identifying leukemia. They described the performance of the techniques, as well as each of their benefits and drawbacks, such as poor robustness. However, there remained a need to improve the accuracy of the existing techniques to develop an automated system. Therefore, in the study reported in this paper, effective deep learning techniques named SegNet- and EfficientNet-based segmentation and feature extraction were conducted. The SegNet architecture improves boundary delineation and can extract highly discriminative features. Subsequently, an XGBoost technique was established to categorize the WBC images into four types: lymphocytes, monocytes, eosinophils, and neutrophils. The major contributions of this research are as follows:

(1) An effective method is proposed for the

segmentation and feature extraction of WBC images using SegNet- and EfficientNet-based networks.

(2) An effective XGBoost-based system is proposed for classifying WBCs into four different classes, namely, neutrophils, eosinophils, monocytes, and lymphocytes.

(3) A detailed investigation is conducted on the publicly available BCCD dataset to create a robust and generalized framework for WBC classification.

The remaining portion of this work is structured as follows. Section 2 discusses the previous research on WBC segmentation and classification. The proposed methodology and its details are given in Section 3. Descriptions of the dataset, performance metrics, results, discussion, and visualization of the proposed approach are included in Section 4. Finally, the conclusions are presented in Section 5.

Proposed Methodology

The proposed approach for segmenting and classifying WBCs into four distinct classes is presented in this section. It contains four main phases, namely, preprocessing, image segmentation, feature extraction, and classification. Initially, the input images are preprocessed to enhance the images of the dataset and then the preprocessed images are segmented using SegNet-based deep learning networks for accurate classification. Then, the segmented image features are extracted from the images using the EfficientNet architecture and classified into the classes of neutrophils, eosinophils, monocytes, and lymphocytes using an XGBoost classifier. These components are discussed in detail in successive sections, and the system architecture of the proposed approach is given in Fig. 1.

Preprocessing

To enhance the training phase of the proposed model, the pictures in the dataset are rescaled into 256 × 256 pixels and converted into grayscale format in the preprocessing stage. To reduce model overfitting, these blood cell images are then subjected to an image augmentation approach. Shearing, flipping, rotation (45°), and brightness improvement are all used to augment the data.

Segnet-based image segmentation

SegNet is a network architecture that is used in the semantic segmentation of an image based on each

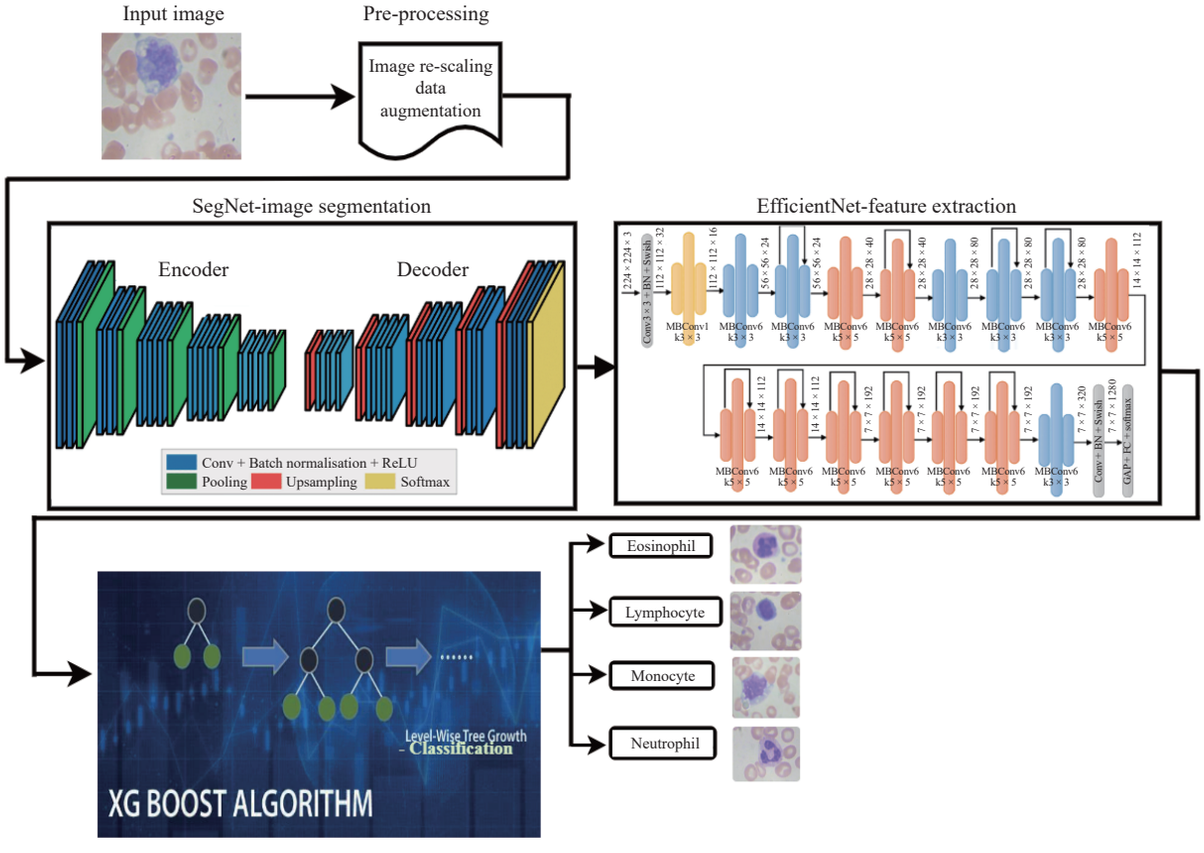

Fig. 1 Proposed framework.

pixel. The network is mostly made up of two parts: encoders and decoders. Rectified linear unit nonlinearity and convolution are the most important components of the encoder part. These are batch-normalized. The last section of the encoder contains the maximum pooling layer. Thirteen convolution layers are included in the encoder. Each encoder layer has its decoder; therefore, the decoder has 13 layers as well. To build a segmentation mask, the decoder's final output is sent into a softmax classifier. To build a collection of feature maps, each encoder performs convolution utilizing filter banks. Subsequently, batch normalization is used. A feature-wise ReLU is then applied as the activation function. After that, the result is down-sampled by a factor of two using a stride 2 non-overlapping window and 2 × 2 max-pooling windows. To achieve shift invariability over lower spatial translations in the image, max-pooling is used. For each feature in the map, down-sampling gives a large number of spatial windows or contexts.

Afterwards, in the decoder section, the feature maps are up-sampled using the pool indices. Then, the feature maps are convolutionally applied by using the filter banks. This process is used to obtain the high-dimensional feature maps. Subsequently, these maps are batch-normalized. Initially, the encoder accepts the input image and the outputs produced by the decoder are multiple pipeline feature maps. The result of the last decoder (high-dimensional feature map) is transferred to the softmax layer. This softmax categorizes and predicts segmentation based on each characteristic. The class with the highest probability at each pixel corresponds to the expected segmentation.

Feature extraction using EfficientNet

The segmented WBC image is given as an input to EfficientNet for feature extraction. In EfficientNet, various mobile inverted bottleneck convolution (MBConv) blocks are included in the baseline architecture with Swish activation function, batch normalization, and squeeze-and-excitation. The sigmoid and linear activation are multiplied together to form the Swish function. EfficientNet consistently scaled all resolution (r), width (w), and depth (d) dimensions using simple and efficient composite coefficients. The number of channels in any layer is referred to as width, while the number of layers in CNN is referred to as depth, and the size of the image is referred to as resolution. Compound scaling is based on the idea that scaling any network dimension (image resolution, depth, and width) can improve accuracy. During the model scaling, the availability of resources is determined by the compound coefficient. This computation is given in the following:

$$d = \alpha^{\phi} \tag{1}$$

$$w = \beta^{\phi} \tag{2}$$

$$r = \gamma^{\phi} \tag{3}$$

$$\text{s.t. } \alpha\beta^2\gamma^2 \approx 2, \alpha \geqslant 1, \beta \geqslant 1, \gamma \geqslant 1$$

Here, the compound coefficient is denoted by ϕ and each dimension's scaling coefficients are α, β, and γ, which can be selected by a grid search. Following determination of the scaling coefficients, the baseline network is scaled to attain the required target model size. For example, When ϕ= 1, the optimal values γ = 1.15, β = 1.1, and α = 1.2 are found using a grid search. EfficientNet-B0 is made up of two residual blocks (Conv) and seven MBConv, which differ in numerous ways, such as in terms of reduction ratio, feature map expansion phase, and kernel size. Depth-wise convolution is used in MBConv6, k3 × 3 and MBConv1, k3 × 3, which includes 3 × 3 kernel size with s stride size. Convolution with 1 × 1 kernel size activation and batch normalization are incorporated in these two blocks. In MBConv6, k3 × 3, a dropout layer and skip connection are included. Although it is not the same as for MBConv1, k3 × 3. Furthermore, MBConv6, k3 × 3 is six times that of MBConv1and k3 × 3 for the extended feature map, and the reduction rate r is fixed to 24 and 4 for MBConv6, k3 × 3 and MBConv1, k3 × 3, respectively. When both MBConv6, k5 × 5 and MBConv6, k3 × 3 conduct the same operations, MBConv6, k5 × 5 has a kernel size of 5 × 5, and MBConv6, k3 × 3 uses a kernel size of 3 × 3. The Adam optimizer was used to speed up the training process instead of random initialization of network weights. Finally, the obtained feature maps are given to the classifier to perform classification.

Classification using XGBoost

For WBC classification, an XGBoost-based classification process is proposed in this section. XGBoost is a machine learning technique that is commonly used for both regression and classification. It is a group of decision trees with gradient boosts.

Initially, a group of K nodes appearing in the ensemble of classification and regression trees (CARTs) were used. For the kth tree, the total prediction scores at a leaf node f_k are used to compute the final prediction result of a class label, which is $\hat{y}_i$ as shown in the equation below:

$$\hat{y}_i = \sum_{k=1}^{K} f_k(x_i), \quad f_k \in F \tag{4}$$

Here, for all CARTs, the set of all K scores is denoted by F and the training set is denoted by x_i. Then, a regularization process is implemented to obtain better outputs, which are expressed as

$$L(\varphi) = \sum_i \ell(\hat{y}_i, y_i.) + \sum_k \Omega(f_k) \tag{5}$$

Here, the error variations among predicted class labels $\hat{y}_i$ and target y_i are calculated to define the differentiable loss function ℓ. In the second portion, to eliminate issue of overfitting, penalization Ω is performed on the model complexity. The penalty Ω function is calculated as follows:

$$\Omega(f) = \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2 \tag{6}$$

Here, to control the regularization degree, the configurable parameters λ and γ are used. The weight values are stored in w_j for each leaf, and leaves are denoted by T. Subsequently, gradient boosting (GB) is used with the loss function to resolve the classification problem, which is then extended by a second Taylor expansion. In step t, the constant term is omitted to obtain a simple target, which is determined as follows:

$$\begin{aligned} \bar{L}^{(t)} &= \sum_{i=1}^{n}\left[g_i f_t(x_i) + \frac{1}{2}h_i f_t^2(x_i)\right] + \Omega(f_t) = \\ &\sum_{i=1}^{n}\left[g_i f_t(x_i) + \frac{1}{2}h_i f_t^2(x_i)\right] + \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2 = \\ &\sum_{j=1}^{T}\left[\left(\sum_{i\in I_j} g_i\right) w_j + \frac{1}{2}\left(\sum_{i\in I_j} h_i + \lambda\right) w_j^2\right] + \gamma T \end{aligned} \tag{7}$$

Here, the instance of leaf t is denoted by $I_j = \{i | q(x_i) = j\}$ and the equations for the loss function' s first j_i and second h_i order gradient statistics are expressed as follows:

$$g_i = \frac{\partial \ell(\hat{y}_i^{t-1}, y_i)}{\partial \hat{y}_i^{t-1}} \tag{8}$$

$$h_i = \frac{\partial^2 \ell(\hat{y}_i^{t-1}, y_i)}{\partial (\hat{y}_i^{t-1})^2} \tag{9}$$

The appropriate weight w_j^* of leaf j is determined as follows:

$$w_j^* = \frac{\sum_{i\in I_j} g_i}{\sum_{i\in I_j} h_i + \lambda} \tag{10}$$

The quality of the tree structure is measured using the scoring function q. For the x_i tree structure, the quality q is determined by the following equation:

$$\bar{L}^{(t)}(q) = -\frac{1}{2}\sum_{j=1} T \frac{(\sum_{i\in I_j} g_i)^2}{\sum_{i\in I_j} h_i + \lambda} + \gamma T \tag{11}$$

To compute the split nodes after splitting, the loss reduction is calculated by applying scores in the instance set of right I_{R} and left I_{L} nodes, which is given as follows:

$$L_{\text{split}} = \frac{1}{2}\left[\frac{(\sum_{i\in I_{\mathrm{L}}} g_i)^2}{\sum_{i\in I_{\mathrm{L}}} h_i + \lambda} + \frac{(\sum_{i\in I_{\mathrm{R}}} g_i)^2}{\sum_{i\in I_{\mathrm{R}}} h_i + \lambda} + \frac{(\sum_{i\in I} g_i)^2}{\sum_{i\in I} h_i + \lambda}\right] - \gamma \tag{12}$$

here, $I = I_{\mathrm{R}} \cup I_{\mathrm{L}}$.

The deep features collected from the training set were used to train the XGBoost classifier. After training the XGBoost model, the deep features extracted from the testing set were used to test the model. Finally, XGBoost predicts which class of WBCs the image should be classified into.

Results and Discussion

This section describes the experiments and details of the dataset, as well as the parameter settings, results, and discussion. All experiments were performed in a system with the Windows 10 operating system, 32 GB of RAM, Nvidia GTX1080Ti GPU, and an Intel Core i7 processor. Keras with the tensor flow backend was used to conduct all of the experiments involving the proposed approach.

Dataset

The data for this study were taken from the BCCD dataset. It was retrieved from https://www.kaggle.com/paultimothymooney/blood-cells/data. It contains 367 photos with a resolution of 640 pixels × 480 pixels. Eosinophils, lymphocytes, monocytes, basophils, and

neutrophils are the five different blood cell classes included in this dataset. Basophil images were excluded (because only five images were included in this class) and the images of the other four kinds of WBCs were used for WBC classification. The blood smears from various patients were examined through the Gismo Right approach using a standard light microscope equipped with a 100× objective lens and the images were captured using an analog CCD color camera to create the dataset. Various picture augmentation approaches were used to enhance the number of images in the dataset, as detailed in the methodology section.

Training and testing

The test set contained 20% of the dataset's images, while the training set contained the remaining 80%, for training and testing the proposed framework. To optimize the adaptive learning process, the Adaptive Moment Estimate (Adam) was used as an optimizer. It generally produces superior outcomes compared with other optimizers, performs quicker computation, and needs less tuning parameters. In XGBoost, the values of hyperparameters such as n-estimators, max. depth, min-child weight, sub-sample, and gamma are 300, 6, 3, 1, and 0, respectively. For segmentation, the accuracy and loss graph of training and testing is shown in Figs. 2(a) and 2(b). Meanwhile, for classification, the accuracy and loss graph of training and testing in is presented in Figs. 3(a) and 3(b).

The segmentation framework was trained with an initial learning rate of 0.001, a batch size of 8, and L2 regularization of 0.0005. The model will fail to recognize significant data structures if the learning rate is inadequate. Meanwhile, errors may occur if it is large. As a result, we decided to set the learning rate to 0.001. Similar to this, a higher batch size necessitates additional memory for training and smaller batch size produce more noise in the error computations. Therefore, the network was configured with a batch size of 8. An L2 regularization and dropout approach was implemented to avoid overfitting. The simulation was carried out for 100 epochs, and the outcomes were examined on the 100th epoch. After 50th epoch, the training loss decreased to its lowest value of 0.8, and the corresponding training accuracy was 91.24%, as depicted in Fig. 2. As shown in Fig. 3, the validation process loss reached its lower limit of 0.75 after 55 training epochs, and the associated validation accuracy was 93.78%. These training charts clarify

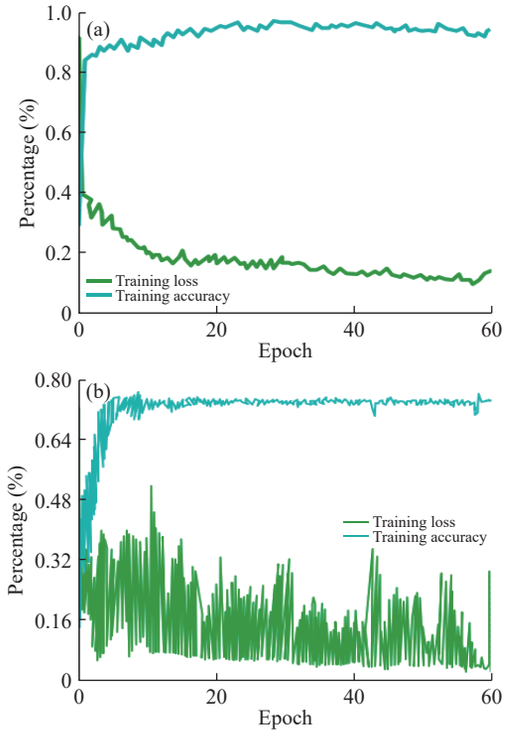

Fig. 2 Accuracy and loss graphs for training and testing in the segmentation framework. **(a)** Accuracy and loss during training. **(b)** Accuracy and loss during testing.

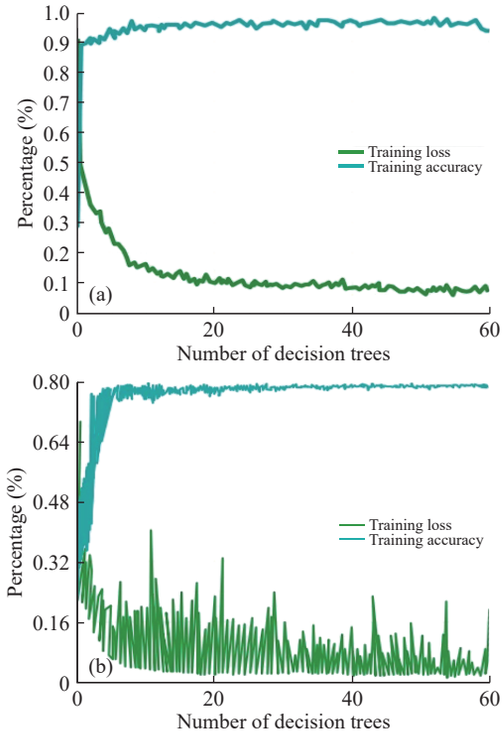

Fig. 3 Accuracy and loss graphs for training and testing in the classification framework. **(a)** Accuracy and loss during training. **(b)** Accuracy and loss during testing.

that the proposed model performed better, increasing classification accuracy while reducing validation loss. Moreover, the testing loss and accuracy plots indicate

that the proposed model was not overfitted with training data in both segmentation and classification phases.

Segmentation results

The suggested WBC segmentation algorithm's performance was measured using three metrics: dice similarity coefficient (DSC), sensitivity, and precision.

$$\mathrm{DSC} = 2 \times \frac{\mathrm{TP}}{(\mathrm{TP} + \mathrm{FP}) + (\mathrm{TP} + \mathrm{FN})} \tag{13}$$

$$\mathrm{Sensitivity} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{14}$$

$$\mathrm{Precision} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \tag{15}$$

The above equations produce these measures using the false negative (FN), true positive (TP), false positive (FP), and true negative (TN) of the final segmentation. Based on these metrics, the obtained quantitative findings of the proposed segmentation strategy are given in Table 1 and Fig. 4.

Table 1 Comparison of WBC cell types for four classes in segmentation

WBC cell type	DSC (%)	Precision (%)	Sensitivity (%)
Neutrophils	98.73	99.84	99.81
Eosinophils	99.23	99.78	99.89
Monocytes	98.84	99.87	99.91
Lymphocytes	98.65	99.72	99.81

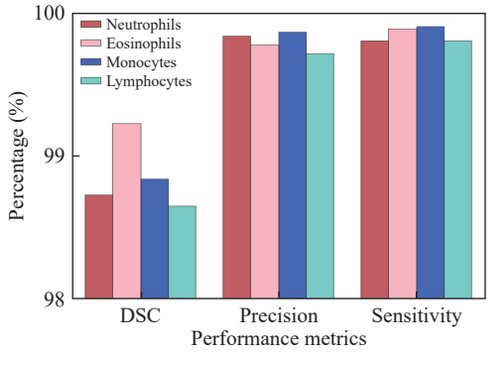

Fig. 4 Performance comparison of WBC cell types in four classes (segmentation).

Table 1 shows that the eosinophil class attains high DSC values, while it has slightly lower values than the other classes in terms of precision and sensitivity. Meanwhile, the precision and sensitivity values of the monocyte class are higher than those of the other three classes, and neutrophils also achieve similar values. Compared with the other classes, the performance of lymphocyte segmentation is slightly worse in terms of all of the applied metrics.

To analyze the performance of the proposed approach, the quantitative results of the proposed technique are evaluated with existing state-of-the-art techniques. These results are given in Table 2 and Fig. 5.

Table 2 Comparison of proposed segmentation technique

Techniques	DSC (%)	Precision (%)	Sensitivity (%)
K-means algorithm [29]	-	94.38	99.78
Otsu's thresholding [26]	96.75	99.72	95.26
VGG-UNet [30]	93.94	89.48	98.87
UNet [30]	94.22	90.80	99.21
Proposed approach	98.86	99.80	99.85

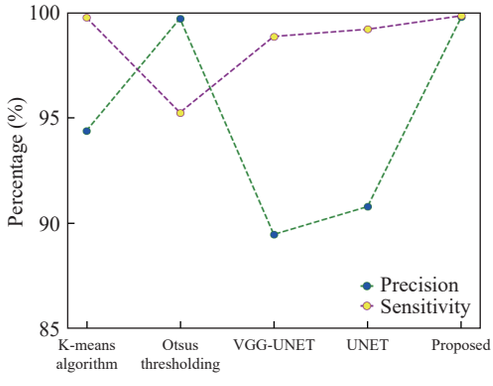

Fig. 5 Comparison of precision and sensitivity for segmentation.

From Table 2, it is observed that the performance of Otsu's thresholding technique is superior to that of all the other techniques, but not to our proposed approach. Moreover, in terms of sensitivity, the K-means algorithm attains a good value (99.78%), which is similar to that of the proposed approach (99.85%). This means that the proposed model has a high capacity to detect the background class of the image.

Finally, the proposed segmentation technique was identified and segments the nucleus with a higher DSC, sensitivity, and precision of 98.86%, 99.85%, and 99.80% respectively.

Classification results

For quantitative assessment of the classification performance, conventional metrics such as F1-score, precision, accuracy, specificity, and sensitivity were used:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}} \quad (16)$$

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (17)$$

$$\text{Specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}} \quad (18)$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (19)$$

$$\text{F1-score} = \frac{2 \times \text{TP}}{2 \times \text{TP} + \text{FP} + \text{FN}} \quad (20)$$

Here, true positive (TP) denotes the number of accurately identified blood cells, and the number of erroneously detected blood cell types is shown by the false-negative (FN) indicator. The number of cells properly recognized as not the correct blood cell type is referred to as true negative (TN), whereas the number of cells wrongly recognized as not the correct blood cell type is referred to as false positive (FP). The confusion matrix for the classification process is displayed in Fig. 6. This matrix shows that the lymphocyte samples were accurately identified at a rate of 99.25%, which is a higher accuracy than that for the other samples. A total of 605 lymphocyte samples were correctly classified from 630 samples (true positives). Meanwhile, few neutrophil, monocyte, and lymphocyte samples were incorrectly classified by the system.

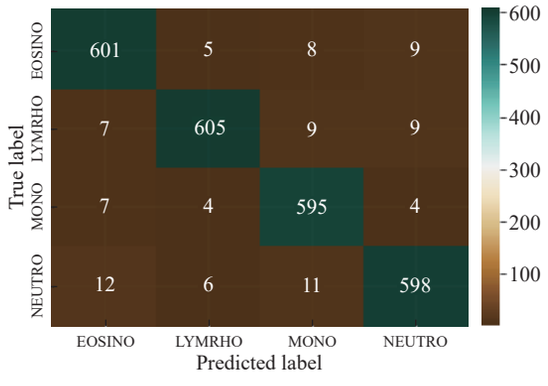

Fig. 6 Confusion matrix for the proposed approach.

To assess the classification performance of the suggested approach, a comparison was performed based on each class, as shown in Table 3 and Fig. 7.

By carefully examining the F1-score metric, which encompasses both sensitivity and precision, it can be concluded that the suggested strategy produced the best results in the majority of the classes. The suggested method correctly identified neutrophils, eosinophils, monocytes, and lymphocytes in the BCCD dataset with F1-scores of 98.86%, 98.78%, 98.38%, and 98.83%, respectively. Among all classes, the classification performance of lymphocytes was better than that for the other classes. The classification accuracy was worse for monocytes than for eosinophils, lymphocytes, and neutrophils.

Table 3 Comparison of WBC cell types in classification

WBC cell type	Accuracy (%)	Precision (%)	Sensitivity (%)	Specificity (%)	F1-score (%)
Neutrophils	98.95	99.12	98.37	99.45	98.86
Eosinophils	99.02	98.62	98.94	98.54	98.78
Monocytes	98.89	98.85	99.14	98.97	98.38
Lymphocytes	99.25	99.36	98.89	99.47	98.83

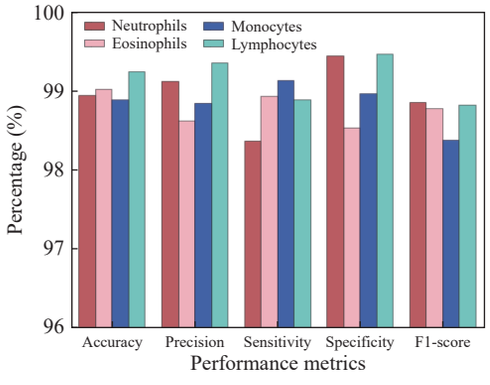

Fig. 7 Performance comparison of WBC cell types in four classes (Classification).

To evaluate the efficiency of the classification process, the proposed approach was compared with existing techniques, the results of which are given in Table 4 and Fig. 8.

The suggested approach had the best performance with precision, sensitivity, and F1-score of 98.98%, 98.83%, and 98.71%, whereas Two-DCNN had the poorest performance in terms of these three variables for the BCCD dataset (Table 4). In the comparison of all approaches, C-SVM and CCA-CCN-RNN outperformed Two-DCNN and Densenet-121, but did not outperform the proposed approach. In Densenet-121 networks, a lack of feature learning led to incorrect labeling. Furthermore, the employed VGG networks were unable to address the issue of overfitting.

The Two-DCNN and Densenet-121 networks performed particularly poorly in WBC classification. This was attributed to the fact that the excessive connections of Densenet-121 produced overfitting, while the Two-DCNN technique always reprocessed

Table 4 Comparison of classification networks

Method	Accuracy (%)	Precision (%)	Sensitivity (%)	Specificity (%)	F1-score (%)
Two-DCNN [21]	—	91.6	91.6	—	91.6
C-SVM [31]	98.44	96.67	96.12	98.04	97.23
CCA-CNN-RNN [27]	—	95.83	95.82	98.43	95.82
Densenet-121 [32]	94	94.22	93.74	—	93.81
Proposed	99.02	98.98	98.83	99.10	98.71

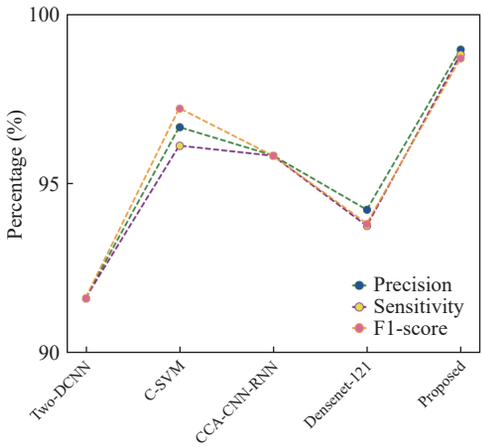

Fig. 8 Comparison of F1-score, sensitivity, and precision metrics for classification.

features from the bottom layers, which exaggerated the divergence of WBC feature extraction. These results indicated that the proposed approach has a superior scalability and resilience in WBC classification.

Conclusion

This study developed a deep learning- and machine learning-based hybrid method for the automatic detection of WBC subsets from peripheral blood smear images. The proposed framework classified the WBCs into four major classes and showed the highest performance for all of these classes. Specifically, lymphocyte, monocyte, eosinophil, and neutrophil cell types were detected with accuracy rates of 99.25%, 98.89%, 99.02%, and 98.95%, respectively. The overall accuracy of our proposed approach was 99.02%. The experiments revealed that the proposed approach outperformed the existing models. Moreover, the proposed model's segmentation performance was also superior to those of all other competitive methods. In the future, we intend to continue to improve the current network architecture, as well as refine and develop an automatic categorization system that can recognize not only WBCs but also other blood cells to help doctors make better diagnoses.

CRediT Author Statement

Bairaboina Sai Samba SivaRao: Conceptualization, methodology, software, formal analysis, investigation, resources, writing original draft, review & editing, and visualization. **Battula Srinivasa Rao**: Conceptualization, writing, review, and editing.

Conflict of Interest

The authors declare that they have no conflict of interests.

References

[1] A. Khan, A. Eker, A. Chefranov, et al. White blood cell type identification using multi-layer convolutional features with an extreme-learning machine. *Biomedical Signal Processing and Control*, 2021, 69: 102932. https://doi.org/10.1016/j.bspc.2021.102932

[2] J. Yao, X. Huang, M. Wei, et al. High-efficiency classification of white blood cells based on object detection. *Journal of Healthcare Engineering*, 2021, 2021: 1615192. https://doi.org/10.1155/2021/1615192

[3] A. Meenakshi, J. A. Ruth, V. R. Kanagavalli, et al. Automatic classification of white blood cells using deep features based convolutional neural network. *Multimedia Tools and Applications*, 2022, 81: 30121−30142. https://doi.org/10.1007/s11042-022-12539-2

[4] J. Pfeil, A. Nechyporenko, M. Frohme, et al. Examination of blood samples using deep learning and mobile microscopy. *BMC Bioinformatics*, 2022, 23(1): 65. https://doi.org/10.1186/s12859-022-04602-4

[5] A. Girdhar, H. Kapur, V. Kumar. Classification of white blood cell using convolution neural network. *Biomedical Signal Processing and Control*, 2022, 71: 103156. https://doi.org/10.1016/j.bspc.2021.103156

[6] Q. Zhai, B. Fan, B. Zhang, et al. Automatic white blood cell classification based on whole-slide images with a deeply aggregated neural network. *Journal of Medical and Biological Engineering*, 2022, 42: 126−137. https://doi.org/10.1007/s40846-022-00683-x

[7] E. Başaran. Classification of white blood cells with SVM by selecting SqueezeNet and LIME properties by mRMR

method. *Signal*, *Image and Video Processing*, 2022, 16: 1821−1829. https://doi.org/10.1007/s11760-022-02141-2

[8] R.A. Bagido, M. Alzahrani, M. Arif. White blood cell types classification using deep learning models. *International Journal of Computer Science & Network Security*, 2021, 21(9): 223−229.

[9] E. Cengil, A. Çınar, M. Yıldırım. A hybrid approach for efficient multi-classification of white blood cells based on transfer learning techniques and traditional machine learning methods. *Concurrency and Computation: Practice and Experience*, 2022, 34(6): e6756. https://doi.org/10.1002/cpe.6756

[10] M. Makem, A. Tiedeu, G. Kom, et al. A robust algorithm for white blood cell nuclei segmentation. *Multimedia Tools and Applications*, 2022, 81(13): 17849−17874. https://doi.org/10.1007/s11042-022-12285-5

[11] Z. Wang, J. Xiao, J. Li, et al. WBC-AMNet: Automatic classification of WBC images using deep feature fusion network based on focalized attention mechanism. *PLoS One*, 2022, 17(1): e0261848. https://doi.org/10.1371/journal.pone.0261848

[12] D. Ryu, J. Kim, D. Lim, et al. Label-free white blood cell classification using refractive index tomography and deep learning. *BME Frontiers*, 2021, 2021(1): 18−26.

[13] N. Alofi, W. Alonezi, W. Alawad. WBC-CNN: Efficient CNN-based models to classify white blood cells subtypes. *International Journal of Online & Biomedical Engineering*, 2021, 17(13): 135−150.

[14] Y. Ha, Z. Du, J. Tian. Fine-grained interactive attention learning for semi-supervised white blood cell classification. *Biomedical Signal Processing and Control*, 2022, 75: 103611. https://doi.org/10.1016/j.bspc.2022.103611

[15] A.C.B. Monteiro, R.P. França, R. Arthur, et al. AI Approach based on deep learning for classification of white blood cells as a for e-healthcare solution. In: Intelligent Interactive Multimedia Systems for e-Healthcare Applications. Singapore: Springer, 2022: 351–373. https://doi.org/10.1007/978-981-16-6542-4_18

[16] Y. Afriyie, B. Weyori, A. AOpoku. Classification of blood cells using optimized capsule networks. *Neural Processing Letters*, 2022, 54: 4809−4828. https://doi.org/10.1007/s11063-022-10833-6

[17] C. Jung, M. Abuhamad, D. Mohaisen, et al. WBC image classification and generative models based on convolutional neural network. *BMC Medical Imaging*, 2022, 22(1): 94. https://doi.org/10.1186/s12880-022-00818-1

[18] G. Hcini, I. Jdey, A. Heni, et al. Hyperparameter optimization in customized convolutional neural network for blood cells classification. *Journal of Theoretical and Applied Information Technology*, 2021, 99(22): 5425−5440.

[19] N. Dong, Q. Feng, M. Zhai, et al. A novel feature fusion based deep learning framework for white blood cell classification. *Journal of Ambient Intelligence and Humanized Computing*, 2022. https://doi.org/10.1007/s12652-021-03642-7

[20] A. Haider, M. Arsalan, Y.W. Lee, et al. Deep features aggregation-based joint segmentation of cytoplasm and nuclei in white blood cells. *IEEE Journal of Biomedical and Health Informatics*, 2022, 26(8): 3685−3696. https://doi.org/10.1109/JBHI.2022.3178765

[21] X. Yao, K. Sun, X. Bu, et al. Classification of white blood cells using weighted optimized deformable convolutional neural networks. *Artificial Cells, Nanomedicine, and Biotechnology*, 2021, 49(1): 147−155. https://doi.org/10.1080/21691401.2021.1879823

[22] C. Cheuque, M. Querales, R. León, et al. An efficient multi-level convolutional neural network approach for white blood cells classification. *Diagnostics*, 2022, 12(2): 248. https://doi.org/10.3390/diagnostics12020248

[23] A. Çınar, S.A. Tuncer. Classification of lymphocytes, monocytes, eosinophils, and neutrophils on white blood cells using hybrid Alexnet-GoogleNet-SVM. *SN Applied Sciences*, 2021, 3: 503. https://doi.org/10.1007/s42452-021-04485-9

[24] Y. Lu, X. Qin, H. Fan, et al. WBC-Net: A white blood cell segmentation network based on UNet++ and ResNet. *Applied Soft Computing*, 2021, 101: 107006. https://doi.org/10.1016/j.asoc.2020.107006

[25] S. Tarek, H.M. Ebied, A.E. Hassanien, et al. White blood cells segmentation and classification using swarm optimization algorithms and multilayer perceptron. *International Journal of Sociotechnology and Knowledge Development*, 2021, 13(2): 16−30. https://doi.org/10.4018/IJSKD.2021040102

[26] S. Tavakoli, A. Ghaffari, Z.M. Kouzehkanan, et al. New segmentation and feature extraction algorithm for classification of white blood cells in peripheral smear images. *Scientific Reports*, 2021, 11(1): 19428. https://doi.org/10.1007/978-3-031-20650-4_13

[27] A.M. Patil, M.D. Patil, G.K. Birajdar. White blood cells image classification using deep learning with canonical correlation analysis. *IRBM*, 2021, 42(5): 378−389. https://doi.org/10.1016/j.irbm.2020.08.005

[28] A. Mittal, S. Dhalla, S. Gupta, et al. Automated analysis of blood smear images for leukemia detection: a comprehensive review. *ACM Computing Surveys*, 2021, 54(11s): 234. https://doi.org/10.1145/3514495

[29] P.P. Banik, R. Saha, K.D. Kim. An automatic nucleus segmentation and CNN model based classification method of white blood cell. *Expert Systems with Applications*, 2020, 149: 113211. https://doi.org/10.1016/j.eswa.2020.113211

[30] S. Kadry, V. Rajinikanth, D. Taniar, et al. Automated segmentation of leukocyte from hematological images—A study using various CNN schemes. *The Journal of Supercomputing*, 2022, 78(5): 6974−6994. https://doi.org/10.1007/s11227-021-04125-4

[31] A. Shahzad, M. Raza, J.H. Shah, et al. Categorizing white blood cells by utilizing deep features of proposed 4B-AdditionNet-based CNN network with ant colony optimization. *Complex & Intelligent Systems*, 2021, 8: 3143−3159. https://doi.org/10.1007/s40747-021-00564-x

[32] F. BOZKURT. Classification of blood cells from blood cell images using dense convolutional network. *Journal of Scientific, Technology and Engineering Research*, 2021, 2(2): 81–88. https://doi.org/10.53525/jster.1014186